Search Results

Search Results for: **[(busy <or> execut% <or> process%) <near> (status <or> indicat% <or> signal% <or> flag% <or> tag)<AND>(((detect% <or> determin% <or> find%) AND (resolv% <or> fix% <or> debug% <or> correct%) <near> path AND circular))]**
Found **519** of **126,861 searched.**

Warning: Maximum result set of 200 exceeded. Consider refining.

Search within Results

1 The implementation of an algorithm to find the convex hull of a set of three-dimensional points 84%

A. M. Day

ACM Transactions on Graphics (TOG) January 1990
Volume 9 Issue 1

A detailed description of the implementation of a three-dimensional convex hull algorithm is given. The problems experienced in the production and testing of a correct and robust implementation of a geometric algorithm are discussed. Attention is paid to those issues that are often brushed over in the theoretical descriptions but cause errors in a real computation. These include degeneracies such as coplanar points, floating-point errors, and other special, but not necessarily degenerate, c ...

2 Program development for a systolic array 84%

Bernd Bruegge

ACM SIGPLAN Notices , Proceedings of the ACM/SIGPLAN conference on Parallel programming: experience with applications, languages and systems January 1988
Volume 23 Issue 9

The primary objective of the Warp programming environment (WPE) is to simplify the use of Warp, a high-performance programmable linear systolic array connected to a general-purpose workstation host. WPE permits the development of distributed applications that access Warp either locally from the host or remotely from a large number of workstations connected to a local area network. Its audience includes the user who calls routines from a library, the programmer who develops new algorithms fo ...

3 Signal design and system operation of Globalstar versus IS-95 CDMA—similarities and differences 82%

Leonard Schiff , A. Chockalingam

Wireless Networks January 2000

Volume 6 Issue 1

The Globalstar^{TM} system provides telephone and data services to and from mobile and fixed users in the area between ± 70 degrees latitude. Connection between user terminals and the PSTN is established through fixed terrestrial gateways via a constellation of low earth orbiting (LEO) satellites. Globalstar uses an extension of the IS‐95 CDMA standard that is used in terrestrial digital cellular systems. The LEO satellite link is ...

4 Compression & aggregation: Model-based compression in wireless ad hoc networks 80%

Milenko Drinic , Darko Kirovski , Miodrag Potkonjak

Proceedings of the first international conference on Embedded networked sensor systems November 2003

We present a technique for compression of shortest paths routing tables for wireless ad hoc networks. The main characteristic of such networks is that geographic location of nodes determines network topology. As opposed to encoding individual node locations, at each node our approach groups the remaining nodes in the network into regions. All shortest paths to nodes in a specific region are routed via the same neighboring node. In this paper, we propose an algorithm for dividing a network field ...

5 A hardware accelerator for speech recognition algorithms 80%

T. S. Anantharaman , R. Bisiani

ACM SIGARCH Computer Architecture News , Proceedings of the 13th annual international symposium on Computer architecture June 1986

Volume 14 Issue 2

This paper describes two custom architectures tailored to a speech recognition beam search algorithm. Both architectures have been simulated using real data and the results of the simulation are presented. The paper also describes the design process of the custom architectures and presents a number of ideas on the automatic design of custom systems for data dependent computations.

6 Engineering a security kernel for Multics 80%

Michael D. Schroeder

Proceedings of the fifth ACM symposium on Operating systems principles November 1975

This paper describes a research project to engineer a security kernel for Multics, a general-purpose, remotely accessed, multiuser computer system. The goals are to identify the minimum mechanism that must be correct to guarantee computer enforcement of desired constraints on information access, to simplify the structure of that minimum mechanism to make verification of correctness by auditing possible, and to demonstrate by test implementation that the security kernel so developed is capab ...

7 Modeling concurrency in parallel debugging 80%

W. Hseush , G. E. Kaiser

ACM SIGPLAN Notices , Proceedings of the second ACM SIGPLAN symposium on

Principles & practice of parallel programming February 1990
Volume 25 Issue 3

We propose a debugging language, Data Path Expressions (DPEs), for modeling the behavior of parallel programs. The debugging paradigm is for the programmer to describe the expected program behavior and for the debugger to compare the actual program behavior during execution to detect program errors. We classify DPEs into five subclasses according to syntactic criteria, and characterize their semantics in terms of a hierarchy of extended Petri Net models. The characterizatio ...

8 An efficient RSVP-mobile IP interworking scheme 80%

Sarantis Paskalis , Alexandros Kaloxylos , Evangelos Zervas , Lazaros Merakos
Mobile Networks and Applications June 2003
Volume 8 Issue 3

During the past years, several attempts have been made to develop functionality for mobility management support and QoS provision in the realm of the IP networks. Since IP was not designed to support such functionality, new protocols have been specified and implemented to tackle these issues. Mobile IP is currently the dominant protocol that allows users to retain connectivity while roaming in IP networks. RSVP (Resource reSerVation Protocol) is a well established protocol for reserving network ...

9 A program debugger for a systolic array: design and implementation 80%

Bernd Bruegge , Thomas Gross
ACM SIGPLAN Notices , Proceedings of the 1988 ACM SIGPLAN and SIGOPS workshop on Parallel and distributed debugging November 1988
Volume 24 Issue 1

The Warp machine consists of a programmable linear systolic array connected to a general-purpose workstation host. Warp can be accessed either locally from this host or remotely from a large number of workstations connected to a local area network. Since the linear arrangement of the cells in the array restricts direct input and output with the host to the boundary cells, a source language debugger is important for program development. The Warp debugger is integrated into the Warp Programmi ...

10 MLSR: a novel routing algorithm for multilayered satellite IP networks 80%

Ian F. Akyildiz , Eylem Ekici , Michael D. Bender
IEEE/ACM Transactions on Networking (TON) June 2002
Volume 10 Issue 3

Several IP-based routing algorithms have been developed for low-Earth orbit (LEO) satellite networks in recent years. The performance of the satellite IP networks can be improved drastically if multiple satellite constellations are used in the architecture. In this work, a multilayered satellite IP network is introduced that consists of LEO, medium-Earth orbit (MEO), and geostationary Earth orbit (GEO) satellites. A new multilayered satellite routing algorithm (MLSR) is developed that calculates ...

11 Exploring and exploiting wire-level pipelining in emerging technologies 80%

Michael Thaddeus Niemier , Peter M. Kogge
ACM SIGARCH Computer Architecture News , Proceedings of the 28th annual international symposium on Computer architecture May 2001
Volume 29 Issue 2

Pipelining is a technique that has long since been considered fundamental by computer architects. However, the world of nanoelectronics is pushing the idea of pipelining to new and lower levels — particularly the device level. How this affects circuits and the relationship between their timing, architecture, and design will be studied in the

context of an inherently self-latching nanotechnology termed Quantum Cellular Automata (QCA). Results indicate that this nanotechnology offers t ...

12 MPI support in the Prism programming environment 80%

Steve Sistare , Erica Dorenkamp , Nick Nevin , Eugene Loh
Proceedings of the 1999 ACM/IEEE conference on Supercomputing (CDROM) January 1999

13 Satellite communications systems move into the twenty-first century 80%

Leonard S. Golding
Wireless Networks February 1998
Volume 4 Issue 2

This paper discusses the evolution of communication satellite systems and communications satellite technology from the 1960's to the 1990's. The paper identifies the key attributes of satellite communications that has driven this evolution and now drives the future directions such systems will take. The paper then discusses the future direction of communication satellite systems including DBS, MSS, FSS and hybrid satellite/terrestrial systems. The paper points to the continued evolution of ...

14 VLIW compilation techniques in a superscalar environment 80%

Kemal Ebcioglu , Randy D. Groves , Ki-Chang Kim , Gabriel M. Silberman , Isaac Ziv
ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1994 conference on Programming language design and implementation June 1994
Volume 29 Issue 6

We describe techniques for converting the intermediate code representation of a given program, as generated by a modern compiler, to another representation which produces the same run-time results, but can run faster on a superscalar machine. The algorithms, based on novel parallelization techniques for Very Long Instruction Word (VLIW) architectures, find and place together independently executable operations that may be far apart in the original code. i.e., they may be se ...

15 Enhanced superscalar hardware: the schedule table 80%

J. K. Pickett , D. G. Meyer
Proceedings of the 1993 ACM/IEEE conference on Supercomputing December 1993

16 Collecting more garbage 80%

Pascal Fradet
ACM SIGPLAN Lisp Pointers , Proceedings of the 1994 ACM conference on LISP and functional programming July 1994
Volume VII Issue 3

We present a method, adapted to polymorphically typed functional languages, to detect and collect more garbage than existing GCs. It can be applied to strict or lazy higher order languages and to several garbage collection schemes. Our GC exploits the information on utility of arguments provided by polymorphic types of functions. It is able to detect garbage that is still referenced from the stack and may collect useless parts of otherwise useful data structures. We show how to partially co ...

17 False Path Elimination in Quasi-Static Scheduling 78%

G. Arrigoni , L. Duchini , C. Passerone , L. Lavagno , Y. Watanabe
Proceedings of the conference on Design, automation and test in Europe March 2002

We have developed a technique to compute a Quasi StaticSchedule of a concurrent specification for the software partitionof an embedded system. Previous work did not takeinto account correlations among run-time values of variables,and therefore tried to find a schedule for all possibleoutcomes of conditional expressions. This is advantageouson one hand, because by abstracting data values one canfind schedules in many cases for an originally undecidableproblem. On the other hand it may lead to exp ...

18 Multiscalar processors 78%

Gurindar S. Sohi , Scott E. Breach , T. N. Vijaykumar

ACM SIGARCH Computer Architecture News , Proceedings of the 22nd annual international symposium on Computer architecture May 1995

Volume 23 Issue 2

Multiscalar processors use a new, aggressive implementation paradigm for extracting large quantities of instruction level parallelism from ordinary high level language programs. A single program is divided into a collection of tasks by a combination of software and hardware. The tasks are distributed to a number of parallel processing units which reside within a processor complex. Each of these units fetches and executes instructions belonging to its assigned task. The appearance of a single log ...

19 MX Fiber Optics Cable Data Network message traffic simulation 77%

Elizabeth Y. S. Kung

The Proceedings of the 16th annual simulation symposium on Simulation March 1983

The proposed MX MPS Fiber Optics Cable Data Network would interconnect over 4600 facilities and cover and area of 12000 to 15000 mi2. One of the important factors in the design of such a large network is the traffic created by the various messages flowing through the network. This paper describes the simulation of the proposed network structure, the different kinds of messages flowing through it, and the message routing mechanism. The performance of the network for diff ...

20 Superscalar design: Cherry: checkpointed early resource recycling in out-of-order microprocessors 77%

José F. Martínez , Jose Renau , Michael C. Huang , Milos Prvulovic , Josep Torrellas

Proceedings of the 35th annual ACM/IEEE international symposium on Microarchitecture November 2002

This paper presents *CHeckpointed Early Resource RecYcling (Cherry)*, a hybrid mode of execution based on ROB and checkpointing that decouples resource recycling and instruction retirement. Resources are recycled early, resulting in a more efficient utilization. Cherry relies on state checkpointing and rollback to service exceptions for instructions whose resources have been recycled. Cherry leverages the ROB to (1) not require in-order execution as a fallback mechanism, (2) allow memory re ...

```
Set      Items   Description
S1     2884795   DETECT? OR DETERMIN? OR RECOGNI? OR INTERROGAT? OR VERIF? -
             OR JUDGE? OR AUTHENTICAT? OR VALIDAT? OR IDENTIF?
S2       45396   FLOW()PATH OR DATA()FLOW
S3      381238   BUSY()INDICATOR OR MODE OR STATUS
S4     1325741   TEST? OR DIAGNOS? OR EVALUAT? OR TRACK? OR ANALYS? OR ANAL-
             YZ?
S5      814359   FUNCTIONING OR EXECUTING OR OPERATING OR PROCESSIONG
S6     3674186   OBJECT? OR EVENT? OR ELEMENT? OR PROGRAM? OR PACKET? OR BL-
             OCK?
S7     3170769   BLOCK? OR STOP? OR DELAY? OR HOLD? OR HALT? OR INTERRUPT? -
             OR RESTRICT?
S8     5028504   EXECUT? OR PROCESS? OR PERFORM?
S9         734   S1 (3N) S2
S10      13631   S3 (3N) S4
S11       1934   S10 AND S5
S12          0   S9 AND S10
S13          0   S9 AND S11
S14        740   S11 AND S6
S15        350   S14 AND S7
S16        206   S15 AND S8
S17          0   S9 AND S16
S18          0   S9 AND ("NOT" OR RE)()S8
S19        940   S9 OR S16
S20         15   S9 AND S3
S21        156   S9 AND S4
S22         58   S21 AND S6
S23         32   S22 AND S7
S24         18   S23 AND S8
S25         45   S20 OR S23 OR S24
S26         11   S25 AND IC=G06F?
File 347:JAPIO Oct 1976-2003/Sep(Updated 040105)
         (c) 2004 JPO & JAPIO
File 350:Derwent WPIX 1963-2004/UD,UM &UP=200407
         (c) 2004  Thomson Derwent
```

26/5/1 (Item 1 from file: 347)
DIALOG(R)File 347:JAPIO

02350942 **Image available**
PROGRAM CHECKING SYSTEM

PUB. NO.: 62-267842 [JP 62267842 A]
PUBLISHED: November 20, 1987 (19871120)
INVENTOR(s): CHIGIRA HIDEKI
NAGAI YOSHIAKI
OTAKI NORIKO
APPLICANT(s): HITACHI LTD [000510] (A Japanese Company or Corporation), JP (Japan)
APPL. NO.: 61-111705 [JP 86111705]
FILED: May 15, 1986 (19860515)
INTL CLASS: [4] **G06F-011/28 ; G06F-009/06**
JAPIO CLASS: 45.1 (INFORMATION PROCESSING -- Arithmetic Sequence Units)
JOURNAL: Section: P, Section No. 699, Vol. 12, No. 151, Pg. 40, May 11, 1988 (19880511)

ABSTRACT

PURPOSE: To improve the synthesizing performance and efficiency of a program of a data type reasoning system by using a chart produced in response to the pattern of a data flow to produce the structure of a each module.

CONSTITUTION: A data flow 92 of the data 89 is checked and the data 88 written to the flow 92 is read. Then the types of both data 89 and 88 are checked. A data flow 91 of the input data 87 is checked for an instruction 83 using the data 88 as an output and the data 86 is read. Then the types of both data 87 and 86 are checked together with the types of both data 85 and 84. If an unmatched part is **detected** , the **data flow** is traced like a check **mode** of data types. Thus the unmatched part is successively corrected. These check and correction actions are repeated until no unmatched part exist any more. Thus a complete program is obtained.

26/5/7 (Item 6 from file: 350)
DIALOG(R)File 350:Derwent WPIX

013674771 **Image available**
WPI Acc No: 2001-158983/200116
XRPX Acc No: N01-115883
Computer implemented method for weakest precondition analysis **of computer** program **, involves identifying set of candidates of computer** program **that are potentially defective via preset data flow** analysis
Patent Assignee: MICROSOFT CORP (MICR-N); MICROSOFT CORP (MICT)
Inventor: HALL M L; MITCHELL C L
Number of Countries: 093 Number of Patents: 003
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
WO 200072153	A1	20001130	WO 2000US14051	A	20000522	200116	B
AU 200050382	A	20001212	AU 200050382	A	20000522	200116	
US 6374368	B1	20020416	US 99316684	A	19990521	200232	

Priority Applications (No Type Date): US 99316684 A 19990521
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
WO 200072153	A1	E	23	G06F-011/36	

Designated States (National): AE AG AL AM AT AU AZ BA BB BG BR BY CA CH CN CR CU CZ DE DK DM DZ EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG UZ VN YU ZA ZW
Designated States (Regional): AT BE CH CY DE DK EA ES FI FR GB GH GM GR IE IT KE LS LU MC MW MZ NL OA PT SD SE SL SZ TZ UG ZW

AU 200050382	A		G06F-011/36	Based on patent WO 200072153

Abstract (Basic): WO 200072153 A1

NOVELTY - The potentially defective set of candidates of computer **program** are **identified** by preset **data flow analysis** . Each set of identified candidates are furthered examined by weakest precondition **analysis** to determine whether set of candidates are actually defective.

DETAILED DESCRIPTION - The intermediate form comprising byte code representation, flow graph representation or tree based representation is generated from source code representation of computer **program** so that identification of set of candidates of computer **program** is conducted based on intermediate form of computer **program** . INDEPENDENT CLAIMS are also included for the following:

(a) computerized system for computer **program analysis** ;

(b) **program** for computer **program analysis**

USE - For weakest precondition **analysis** of computer **programs** .

ADVANTAGE - Realizes faster **analysis** since weakest precondition **analysis** is only **performed** on those candidates that are deemed potentially defective by initial predetermined data flow **analysis** .

DESCRIPTION OF DRAWING(S) - The figure shows the **block** diagram of method of **analyzing** .

pp; 23 DwgNo 2/3

Title Terms: COMPUTER; IMPLEMENT; METHOD; WEAK; **ANALYSE** ; COMPUTER; **PROGRAM** ; IDENTIFY; SET; CANDIDATE; COMPUTER; **PROGRAM** ; POTENTIALLY; DEFECT; PRESET; DATA; FLOW; **ANALYSE**

Derwent Class: T01

International Patent Class (Main): **G06F-011/26** ; **G06F-011/36**

International Patent Class (Additional): **G06F-009/45**

File Segment: EPI

26/5/8 (Item 7 from file: 350)

DIALOG(R)File 350:Derwent WPIX

008345824 **Image available**

WPI Acc No: 1990-232825/199031

XRPX Acc No: N90-180569

Digital system memory interface controller - has device for receiving incoming pixel data and for detecting pixel address collision in graphics display system

Patent Assignee: HONEYWELL INC (HONE)

Inventor: FISCHER D A; GRAVES J A; SNODGRASS T D; WOODS J W

Number of Countries: 005 Number of Patents: 005

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
EP 379983	A	19900801	EP 90101055	A	19900119	199031	B
US 4970636	A	19901113	US 89299794	A	19890123	199048	
EP 379983	A3	19930407	EP 90101055	A	19900119	199351	
EP 379983	B1	19970312	EP 90101055	A	19900119	199715	
DE 69030127	E	19970417	DE 630127	A	19900119	199721	
			EP 90101055	A	19900119		

Priority Applications (No Type Date): US 89299794 A 19890123

Cited Patents: NoSR.Pub; 1.Jnl.Ref

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
EP 379983	A				
	Designated States (Regional): DE FR GB IT				
EP 379983	B1	E	49	G06T-015/10	
	Designated States (Regional): DE FR GB IT				
DE 69030127	E			G06T-015/10	Based on patent EP 379983

Abstract (Basic): EP 379983 A

The MIC receives data and a data valid bit set active low. Using the **HALT** bit (set active high) the MIC controls the rate at which the MIC pipeline and GIPP is running. If the bit is low the GIPP output

data is loaded into [redacted] MIC on the next rising edge o[redacted]he clock causing the GIPP data and the MIC to move one register.

When the **HALT** is high MIC and GIPP **hold** data, though typically **HALT** is low every other clock cycle. Whenever the microprocessor control section commands a **HOLD** or when an address collision occurs in the pipeline **HALT** goes high for more than one clock cycle. Whenever a screen clip of a Z clip occurs **HALT** goes low for more than one clock cycle. Both occurrences allowing for a higher effective throughput.

ADVANTAGE - Increases effective memory bandwidth while minimising memory interface and control hardware, eliminates external memory initialisation and self **test** hardware.

Dwg.6/13

Title Terms: DIGITAL; SYSTEM; MEMORY; INTERFACE; CONTROL; DEVICE; RECEIVE; INCOMING; PIXEL; DATA; DETECT; PIXEL; ADDRESS; COLLIDE; GRAPHIC; DISPLAY; SYSTEM
Derwent Class: T01
International Patent Class (Main): G06T-015/10
International Patent Class (Additional): **G06F-015/72**
File Segment: EPI

26/5/9 (Item 8 from file: 350)
DIALOG(R)File 350:Derwent WPIX

007134858
WPI Acc No: 1987-134855/198719
XRPX Acc No: N87-100858

Microprocessors systems digital monitoring appts. - has triggering code detector with output to address and control rails of data processing unit

Patent Assignee: ANDREEV B M (ANDR-I)
Inventor: LEUKHIN S P
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
SU 1259267	A	19860923	SU 3827367	A	19841218	198719	B

Priority Applications (No Type Date): SU 3827367 A 19841218
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
SU 1259267	A		10		

Abstract (Basic): SU 1259267 A

The monitor contg. a data input (1) to the buffer register (2) at an input to the memory (3), buffer memory (6), data processor (7) and the display (11), has a triggering code detector (4) with the address (8), data (9) and control (10) rails. The recording control circuit (5) and the parameter input unit (12) can be reconstructed.

The recording **mode** is set by the data processor and implemented by a specialised recording control circuit. A triggering code word is **detected** in the input **data flow** . The recording and display duty can be changed at a keyboard. Recording can be triggered by a code word, or after detection of n code words. The data preceding a code word over a set number of intervals, or the data for comparison of data-blocks, can be recorded. Data can also be recorded in response to external synchronisation. The search for code words is automatic.

USE/ADVANTAGE - In monitoring complex digital appts. and microprocessor systems in computer engineering, operating speed is increased. Bul.35/23.9.86. (10pp Dwg.No.1/4

Title Terms: MICROPROCESSOR; SYSTEM; DIGITAL; MONITOR; APPARATUS; TRIGGER; CODE; DETECT; OUTPUT; ADDRESS; CONTROL; RAIL; DATA; PROCESS; UNIT
Derwent Class: T01
International Patent Class (Additional): **G06F-011/00**
File Segment: EPI

26/5/10 (Item 9 from file: 350)

DIALOG(R)File 350:Derwent WPIX

004821522
WPI Acc No: 1986-324863/198649
XRPX Acc No: N86-242238
Sampled- data flow **external number** determinator **- uses lower and upper limit inputs to search for numbers at or below lower limit or between limits**
Patent Assignee: KARASEV I A (KARA-I)
Inventor: KORNEEV I L; KUKHNIKOV V I
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
SU 1226441	A	19860423	SU 3769902	A	19840705	198649	B

Priority Applications (No Type Date): SU 3769902 A 19840705
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
SU 1226441	A		5		

Abstract (Basic): SU 1226441 A
The circuitry contg. an n-input OR-gate (7) and as many (n) **analysis** channels (1) as numbers to be compared and each with trigger (2), AND-gates (4,6), NOT-gate (5) and comparison number (13) result readout (28) and **blocking** (31) inputs, has the start (3) and sync. (8) inputs, triggers (9,14,17,21), AND-gates (10, 15,18,22) lower limit (11) and upper limit (20) inputs, NOT-gates (12,16,19) NAND-gates (23-26) multiplexor (27), NOR-gates (29,30,32) XOR-gate (34) and the type of extremum setting input (35).
Numbers arriving high order digits first are compared to find the max. or min. value as a function of the set conditions while observing boundary conditions. In each clock interval the AND-gates implement four functions in terms of the direct and inverse values of corresp. number digits and range boundaries. After as many cycles as digits of the compared numbers, the results of six functions are fed to the multiplexor. The NOR-gates enable priority to be given to the channel with the lowest number if several numbers satisfy a selected function.
USE/ADVANTAGE - In slice sorting of sampled-data flows in computing provision is made to determine numbers within set boundaries.
Bul.15/23.4.86 (5pp Dwg.No.1/1
Title Terms: SAMPLE; DATA; FLOW; EXTERNAL; NUMBER; DETERMINATE; LOWER; UPPER; LIMIT; INPUT; SEARCH; NUMBER; BELOW; LOWER; LIMIT; LIMIT
Derwent Class: T01
International Patent Class (Additional): **G06F-007/02**
File Segment: EPI

26/5/11 (Item 10 from file: 350)
DIALOG(R)File 350:Derwent WPIX

004461126
WPI Acc No: 1985-288004/198546
XRPX Acc No: N85-214553
Digital data block **error detection system - inhibits data flow in faulty** block **is detected and indicates** block **number to user**
Patent Assignee: PIONEER ELECTRONIC CORP (PIOE)
Inventor: JINGUJI T
Number of Countries: 003 Number of Patents: 003
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
NL 8500937	A	19851016	NL 85500937	A	19850329	198546	B
JP 60206390	A	19851017				198548	
US 4847840	A	19890711	US 8796760	A	19870909	198935	

Priority Applications (No Type Date): JP 8462618 A 19840330
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
NL 8500937	A		62		

Abstract (Basic): NL 8500937 A

When the system is set (S1) to the debug **mode** by an external control, the data flow is **tested** (S2) for errors. If a faulty **block** of data is not **detected** (S2N), **data flow** continues (S6). If a faulty **block** is detected (S2Y), the flow is inhibited (S3) and the **block** number is indicated (S4) to the user, who decides (S5) whether to allow the faulty **block** through or not.

If the faulty **block** is allowed through (S5Y), data flow continues (S6) and a check (S7) is made to **determine** if the **data flow** has ended. If it has ended (S7Y), the Debug **mode** is de-selected, if not (S7N), the system checks (S1) whether Debug **mode** is still selected and allows the next **block** of data to be **tested** (S2).

USE - For video disc player with still picture with sound facility.

15/33

Title Terms: DIGITAL; DATA; **BLOCK** ; ERROR; DETECT; SYSTEM; INHIBIT; DATA; FLOW; FAULT; **BLOCK** ; DETECT; INDICATE; **BLOCK** ; NUMBER; USER

Derwent Class: P85; T01; T03; W04

International Patent Class (Additional): **G06F-011/10** ; **G06F-015/40** ; G09B-007/00; G11B-020/18; H03M-013/00; H04N-005/85

File Segment: EPI; EngPI